

## Secrets of the DTS API

- Get versus Find
- Use CASDs
- Refresh with Find
- Nothing changes the database ...
- Update after changes
- Avoid passing the connection

Apelon

---

---

---

---

---

---

---

## Get versus Find

- `getXXX()` – gets the values of system elements or fixed attributes. Few arguments.  
`getAllSynonymTypes()`  
`getName()`
- `findXXX()` – searches the database for information. Multiple arguments/forms.  
`findConceptsWithNameMatching(...)`
- `getFetchedXXX(...)` – returns the attributes in the *client* copy of an object  
`getFetchedConceptAssociations()`

Apelon

---

---

---

---

---

---

---

## Use CASDs

- CASDs let you control the amount of information returned from the database in a query - usually a `findXXX(...)`  
`casd.setAllPropertyTypes()`  
`findConceptsWithNameMatching("a*", ... casd)`
- Use `getFetchedXXX` to see the returned attributes  
`Concept.getFetchedProperties()`
- Be aware of time/space considerations in large finds

Apelon

---

---

---

---

---

---

---

## Refresh With Find

- It is sometimes necessary to “refresh” the attributes in a concept
- Use a findXXXById method for this:

```
//set up new_casd  
con = findConceptById(con.getId(),  
    con.getNamespaceId(), new_casd);
```

Apelon

---

---

---

---

---

---

---

---

## Nothing Changes the Database But Queries

- You must use an XXXQuery class to make changes to the DTS database
- Don't be fooled by:

```
concept.addProperty(new_prop)
```

this only changes the local copy

- You need to do:

```
concept = query.addProperty(concept, new_prop)
```

where query is an instance of DTSConceptQuery

Apelon

---

---

---

---

---

---

---

---

## Update After Changes

- Note the update of your local concept in:

```
concept = query.addProperty(concept, new_prop)
```

- Use the return value; this is necessary to keep the local (client) copy in sync

Apelon

---

---

---

---

---

---

---

---

## Avoid Passing the Connection

- Large applications need frequent access to the DTS Connection Object
- Passing this between classes/methods is inefficient
- Can use a static DTS object for shared access:

```
DTSAppManager.getQuery().setConnection(conn);  
DTSAppManager.getQuery().getSearchQuery() ...
```

- DTSAppManager also facilitates use of JDBC connections

Apelon

---

---

---

---

---

---

---

---